Load the dataset auto.csv (given in the shared Google folder) into R. This dataset has some columns:

**name**: name of the car model (e.g., bmw 2002)

**origin**: where the car is produced. 1 = US, 2 = EU, 3 = Asia.

**mpg**: miles per gallon (the higher the better).

**weight**: weight in lbs

**model\_year**: the model year

**horsepower**: the engine power (measured in horsepower). The higher the stronger.

**cylinders**: the number of cylinders in the car engine. The higher the stronger.

**acceleration**: the time (in seconds) for the car to speed up from 0 to 60 mph.

**displacement**: the volume of air the car engine can take in to burn.

To answer the below questions, we need to set the directory where the “auto.csv” file is located.

In my local machine, the code to set the working directory is

setwd(“C:/Users/cscha/OneDrive/Desktop/Software analytics”)

Now read the “auto.csv” file as typing the following code

> cars <- read.csv(“auto.csv”)

Here, “cars” work as a variable to perform operations on the “auto.csv” file

Q1. Calculate and explain the (pairwise) correlation among cylinders, displacement, horsepower, and weight. Why they have nearly perfect correlation?

In general, correlation is mainly used to compute the strength and direction which are the indicators of a linear relationship between the two variables.

The ’R’ code to calculate the correlation among cylinders, displacement, horsepower, and weight is as follows

> required\_data <- cars[,c("cylinders", "displacement", "horsepower", "weight")]

> output\_data <- cor(required\_data)

> print(output\_data)

Here is the first line of code which is > required\_data <- cars[,c("cylinders", "displacement", "horsepower", "weight")]

subsets the data to include only the columns cylinders, displacement, horsepower, and weight from the “auto.csv” file.

After that, > output\_data <- cor(required\_data), this line of code uses the cor() to determine the pairwise correlations between the selected columns and then stores the correlation matrix.

>print(output\_data) Prints the correlation matrix.

The output of the above code is:

cylinders displacement horsepower weight

cylinders 1.0000000 0.9508233 0.8429834 0.8975273

displacement 0.9508233 1.0000000 0.8972570 0.9329944

horsepower 0.8429834 0.8972570 1.0000000 0.8645377

weight 0.8975273 0.9329944 0.8645377 1.0000000

When we consider the diagonal matrix elements from the above matrix, they always represent the correlation of each variable with itself, which is always 1.00 because the variable is always perfectly correlated with itself.

Considering the correlation between cylinders and displacement:

* Cylinders and displacement have a correlation coefficient of almost 0.9507, which is extremely close to 1 which states that they have a strong linear relationship between them.
* In general, as the number of cylinders increases the displacement also increases which is a common principle behind the car design.

Considering the correlation between cylinders and horsepower:

* Cylinders and horsepower have a correlation coefficient of almost 0.842. This states that they have a moderate to strong linear relationship between them.

Considering the correlation between cylinders and weight:

* Cylinders and weight have a correlation coefficient of almost 0.8975. This states that they have a strong linear relationship between them.
* In general, as larger engines can produce the necessary power to move the greater weight, heavier cars frequently have more cylinders.

Considering the correlation between displacement and horsepower:

* Displacement and horsepower have a correlation coefficient of almost 0.89752. This states that they have a strong linear relationship between them.
* When calculating an engine's power output, engine displacement is a key consideration. In general, higher horsepower results from a larger displacement since it allows for greater air and fuel intake.

Considering the correlation between displacement and weight:

* Displacement and weight have a correlation coefficient of almost 0.9329. This states that they have a strong linear relationship between them.
* In normal terms, larger engines with more displacement are frequently seen in heavier cars to supply the necessary power for their mass, and to run the cars efficiently.

Considering the correlation between horsepower and weight:

* Horsepower and weight have a correlation coefficient of almost 0.8645. This states that they have a strong linear relationship between them.
* We can see the strong linear relationship between them because bigger vehicles require more horsepower to accelerate and move, and they often have stronger engines.

By considering the above results we can state that the correlation coefficient is strong and cars with heavier engines (more cylinders, larger displacement, and higher horsepower) are more prevalent.

Q2. Calculate and explain the correlation of acceleration andhorsepower. Why their correlation is less perfect?

The ‘R’ code to calculate the correlation of acceleration and horsepower is as follows:

> correlation\_horsepower\_acceleration <- cor(cars$horsepower, cars$acceleration)

> print(correlation\_horsepower\_acceleration)

In the above code > correlation\_horsepower\_acceleration <- cor(cars$horsepower, cars$acceleration), it subsets the data to include only the columns horsepower and acceleration from the “auto.csv” file and uses the cor() to determine the correlations between the selected columns.

> print(correlation\_horsepower\_acceleration), it prints the output correlation value.

The output correlation value is:

[1] -0.6891955

The degree and direction of the linear relationship between acceleration and horsepower are quantified by the correlation coefficient between these two parameters. Between -1 and 1, the correlation coefficient can lie. From the above value, we can state that the linear relation between horsepower and acceleration is not strong. The factors that might cause less correlation between horsepower and acceleration are car design and the cars with more powerful engines can experience occasional decreases in acceleration which is caused by the car’s weight.

Q3. model\_year has a pretty good correlation with mpg. What does this suggest? It has a negative (and pretty small) correlation to the group of cylinders, displacement, horsepower, and weight. What does this suggest?

The ‘R’ code to calculate the coefficient correlation between model year and mpg is as follows:

> mpg\_modelyear\_cor <- cor(cars$mpg, cars$model\_year)

> print(mpg\_modelyear\_cor)

In the above code > mpg\_modelyear\_cor <- cor(cars$ mpg, cars$model\_year), it subsets the data to include only the columns mpg and model\_year from the “auto.csv” file and uses the cor() to determine the correlations between the selected columns.

> print(mpg\_modelyear\_cor)), it prints the output correlation value.

The output correlation value is: [1] 0.580541

From the above result, we can state that the correlation between model\_year and mpg is a significant relationship between them. The relationship between a car's model year and its fuel economy (miles per gallon) is shown by the correlation coefficient between "model\_year" and "mpg". A positive correlation indicates that newer model years typically have better fuel efficiency.

The ‘R’ code to calculate the correlation between model\_year and to the group of cylinders, displacement, horsepower, and weight is as follows:

> modelyear\_and\_group\_cor <- cor(cars$model\_year, cars[, c("cylinders", "displacement", "horsepower", "weight")])

> print(modelyear\_and\_group\_cor)

In the above code modelyear\_and\_group\_cor <- cor(cars$model\_year, cars[, c("cylinders", "displacement", "horsepower","weight")]), it binds the model\_year from the “auto.csv” file and gets the bonded subset of cylinders", "displacement", "horsepower","weight" and calculate the correlation between model\_year and this bonded data and stores the correlation matrix in “modelyear\_and\_group\_cor” variable.

The output is:

cylinders displacement horsepower weight

[1,] -0.3456474 -0.3698552 -0.4163615 -0.3091199

Here from the above table, we can state that the correlation between model\_year and the group of "cylinders", "displacement", "horsepower", and "weight” is negative which means the correlation is not strongly linear. It says that

negative correlation between model\_year and another group of car variables suggests that cars tend to have engines with fewer cylinders, lower displacement, less horsepower, and lower weight.

So from the output we state

model\_year and cylinders have a correlation of -0.3456474

model\_year and displacement have a correlation of -0.3698552

model\_year and horsepower have a correlation of -0.4163615

model\_year and weight have a correlation of -0.3091199.

Q4. Explain why it does not make sense to analyze the correlation of origin to other variables like mpg or horsepower. If we want to analyze if origin does relate to mpg or horsepower, what should we do?

From the “auto.csv” file we can say that origin is a categorical nominal variable and mpg and horsepower are

continuous numeric variables, it is not appropriate to compute the correlation between the origin variables and other variables like mpg horsepower because of their characteristic numeric values. The degree and direction of linear correlations between two continuous numeric values such as between mpg and horsepower are measured using correlation coefficients such as Pearson’s correlation coefficient.

Factors that say that it is not appropriate to compute the correlation between continuous numeric and non-continuous numeric values are:

Here origin which is a noncontinuous numeric variable represents values such as 1,2,3 which states that origin value 1= US, origin 2=EU, and origin 3=Asia. In contrast, "mpg" and "horsepower" are continuous variables that have a large range of values. This kind of heterogeneous data is not intended to be handled by correlation coefficients.

Moreover, the origin is a nominal variable that has no particular numerical significance.

To compute the correlation between mpg and horsepower and weight we can perform Analysis of Variance(ANOVA).

ANOVA examines whether there is a statistically significant difference between the means of these continuous variables and the categories of the categorical variable (origin).

Q5. US zip codes are also stored as numbers (5 digits). Explain why it does not make sense to analyze the correlation of zipcode to other variables like house price or household income...

If you want to compare house prices between Lubbock, TX, and Dallas, TX (different zip codes), what should you do?

Here the US zipcode is stored in numbers and they are nominal data that represent the categories without any inherent order, and they don’t have continuous nominal data unlike household prices and household income.

For instance, the zip code format in the US be like ‘79425’ or ‘85476’ whereas household price might be in the form of 23445,24354.65 and household price might be in the form of 5654,3424.56. So the household price and household income are continuous numeric values and zipcode is a non-continuous variable, so correlation cannot be computed in an efficient way.

As, there are some methods to determine the correlation between continuous and non-continuous variables such as ANOVA testing and t-testing can be used to compare house prices between Lubbock, TX and Dallas, TX (different zip codes),

For example, let us consider the below ‘R’ code to compare house prices between Lubbock, TX and Dallas, TX (different zip codes),

> data <- data.frame(

+ zipcode = c("79415", "75201", "79402", "75202", "79403"),

+ houseprice = c(250000, 450000, 220000, 480000, 210000)

+ )

> data\_of\_lubbock <- data[data$zipcode %in% c("79415", "79402"), ]

> data\_of\_dallas <- data[data$zipcode %in% c("75201", "75202"), ]

>

> if (nrow(data\_of\_lubbock) > 0 && nrow(data\_of\_dallas) > 0) {

+ summary(data\_of\_lubbock$houseprice)

+ summary(data\_of\_dallas$houseprice)

+ t\_test\_result <- t.test(data\_of\_lubbock$houseprice, data\_of\_dallas$houseprice)

+ print(t\_test\_result)

+ } else {

+ cat("No data found for Lubbock or Dallas.")

+ }

The output is:

Welch Two Sample t-test

data: data\_of\_lubbock$houseprice and data\_of\_dallas$houseprice

t = -10.842, df = 2, p-value = 0.0084

alternative hypothesis: true difference in means is not equal to 0

95 percent confidence interval:

-321273 -138727

sample estimates:

mean of x mean of y

235000 465000

The above ‘R’ code is s t-test which is a statistical test used to compare the means of two separate samples, in this example, Lubbock and Dallas's home prices. Let's examine the output in detail and describe each component:

Here in the above ‘R’ code data\_of\_lubbock$houseprice and data\_of\_dallas$houseprice are the two datasets that are going to be compared.

Here data\_of\_dallas$houseprice contains house prices in Dallas while data\_of\_lubbock$houseprice contains the house prices in Lubbock.

Here in the above ‘R’ code the t-value evaluates the degree to which the means of the two groups differ in relation to the variation within each group. The t-value for this situation is -10.842.

Degrees of freedom(df) are used to describe how many values in a statistic's final calculation are subject to change. The degrees of freedom in a Welch t-test are not usually a whole number. The df in this instance is 2.

The amount of evidence contradicting a null hypothesis is indicated by the p-value. It provides you with the likelihood of seeing these extreme results, or ones even more severe if the null hypothesis were to be correct. The p-value for this situation is 0.0084 and the alternative hypothesis states what we are testing for.

95 percent confidence interval states that the true difference in mean house prices between Lubbock and Dallas falls within the given range.

In conclusion, the findings of the t-test indicate that the mean housing prices in Lubbock and Dallas are significantly different from one another. House prices in Lubbock are, on average, substantially lower than those in Dallas, according to the negative t-value and the negative confidence interval values. The evidence of a substantial difference is further supported by the low p-value.